



TITLE:

GAを用いた時系列基本パターンの認識: 株価への応用 (不確実・不確定性のもとでの数理的決定理論)

AUTHOR(S):

池田, 欽一

---

CITATION:

池田, 欽一. GAを用いた時系列基本パターンの認識: 株価への応用 (不確実・不確定性のもとでの数理的決定理論). 数理解析研究所講究録 2000, 1132: 198-206

ISSUE DATE:

2000-02

URL:

<http://hdl.handle.net/2433/63699>

RIGHT:

# GA を用いた時系列基本パターンの認識 — 株価への応用 —

九州大学経済学部 池田 欽一 (Yoshikadu IKEDA)

## 1 まえがき

本論文では、取引ルール集合を遺伝的アルゴリズムを用いて最適化する応用的ルール生成システムの構成を試み、株取引シミュレーションを行なった。

人工知能の分野では効率的で、外部環境に適応する知識ベースを構築するクラシファイアシステムについての研究がおこなわれてきた。[1],[2],[9],[10]。本論文では、条件となる部分には過去の株価の変動パターンを与え、行動、結論部分は、条件部のパターンに対する取引内容(売り買い)であると仮定をおくことにより、外部環境(株価市場)に適合した株価変動パターン集合の自動生成システムを考える。

まず、2章ではクラシファイアシステムとはどのようなものであるか、また、その概念を用いてどのような株価投資支援システム(適応的ルール生成システム)を構築するのかを述べる。

3章では、過去の株価変動とそれに対する取引内容のペアで表されるルールを、どのように更新していくかを示す。ルール更新には遺伝的アルゴリズム(GA)を用いる。遺伝的アルゴリズムでは、染色体と呼ばれる遺伝子列によって解候補を表し、適応度と呼ばれる評価によって次世代構成へ寄与する割合をきめ、解候補集団全体の最適値への改良を行なうものである。本システムでは取引ルールを個体集団であるとみなし、取引ルールが外部環境(株価変動)にどの程度あっているかを適応度として効率的ルール集合を求めることになる。

4章では、本システムを日経平均株価の取引戦略に適用した場合の利益率についてのシミュレーション分析を行なった。なお、比較のため、AR モデルによる取引の利益率についてもまとめている。

## 2 適応的ルール生成システム

### 2.1 クラシファイアシステム

クラシファイアシステムとは、*if ~ then* 形式のルールの集合であり、外部環境からの情報入力(の刺激)により、ある出力を行なうシステムである。

クラシファイアシステムは、もともと遺伝的アルゴリズムを用いて、機械学習を行なうものであり、遺伝子空間上にコーディングされたプログラムを遺伝的アルゴリズムを用いて外部環境に適したものを自動的に生成するためのシステムである。

クラシファイアシステムが対象としている環境は、(1) ある一定数のルールで規定できるような安定した環境でない、(2) ノイズの多い環境、(3) 最終的な目標が明確でない環境、(4) 出力値に対する一定の評価基準があり、それによって報酬が与えられる環境を前提としている。

株価変動、投資を考えると一般的に非定常であり、さまざまなノイズをもち、はっきりとした目標値はなく、どれだけの収益があったかという評価を簡単に行なうことができるので、前記の環境の条件を満たしているようである。

クラシファイアシステムは以下の3つの機能から構成されている。

- プロダクションシステム
- 報酬の分配, 信頼度の伝搬
- 遺伝的アルゴリズムによる新ルール生成システム

## 2.2 適応的ルール生成システム

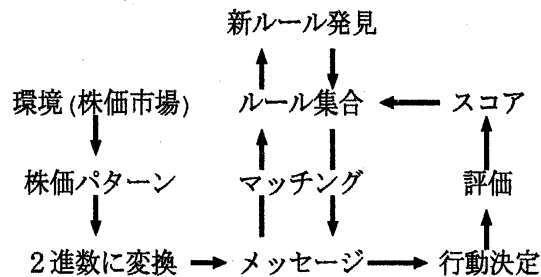


図 1: 適応的ルール生成システムの構成

プロダクションシステムとは、人工知能分野において推論、判断を行なうモデルである。プロダクションシステムは、コンディション (if) 部、アクション (then) 部の対で表現されるプロダクションルール、プロダクションルールの適応対象となるデータを記録するワーキングメモリ、ワーキングメモリの内容とプロダクションルールのコンディション部のマッチングを行ない入力情報に適合するルールを検索し、そのルールのアクション部に従って外部環境に情報を出力するインタプリタによって構成される。

クラシファイアシステムにおけるプロダクションシステムは、インタプリタに対応する入力インターフェイス、分類リスト、出力インターフェイス、ワーキングメモリに対応するメッセージリスト、そして知識を表すルールからなっている。

メッセージとは外部入力や、内部状態を表す記号列で、アルファベットや、0,1 のバイナリ値によって表現する。適応的ルール生成システムでは、取引戦略決定に用いる過去の株価変動パターン、それに対する取引内容がメッセージに対応しており、システム内部では 10 進数表記された値を 2 進数に変換して用いることになる。例えば、0 ~ 1 の間の値をとるように基準化された株価が 0.549 となった場合、0,1 によるビット表現 (10 ビット) は

$$0.549 \rightarrow 0100011001. \quad (1)$$

となる。ルールはメッセージと同様に記号列で表されるコンディション部、アクション部をもち、そのルールの外部環境に対する適応度を表す信頼度という値をもっている。

入力インターフェイスでは、外部からの入力情報を適切な記号列に変換し、メッセージリストに書き込みを行なう。

分類リストは、メッセージリストに書き込まれたメッセージとルールのコンディション部とのマッチングを行ない、条件を満たすルールを検索する。マッチしたルールの内でもっとも信頼度の高いルールが採用され、そのルールのアクション部の記号列が出力インターフェイスによって外部環境への出力とされる。

その後、出力インターフェイスによって出力された値が外部環境に適合しているかどうか評価をおこなう。その評価に従って出力を行なったルールに対して報酬が与えられる。

以上のようなサイクルを一定期間繰り返した後、ルールを生物集団、ルールの信頼度を適応度とする遺伝的アルゴリズムによって新たなルール集合を生成してより外部環境に適したルールの集合を構成していく。

適応的ルール生成システムでは、次の仮定、前提をしておく。ルールの表現方法として、条件部には基本変動パターンを与え、それに対応する結論、行動部には過去の株価変動パターンに対する取引アクションを持っている。(図2参照)。また、ルールには、外部環境に対する信頼値、評価値を与えなければならない。この

### ルール集合

条件部 (パターン)	行動部 (売買シグナル)
0100100101110	0
0010110100100	1
1010010010001	0
⋮	⋮
⋮	⋮
⋮	⋮
⋮	⋮

図 2: ルールを表現する記号列の例

場合のルールの評価基準としては外部環境より過去の株価変動のパターンが示され、それにマッチした条件部をもつルールが活性化し、その結論部 (売り買い) を外部環境に対して実行し、実測値が観測された時に、活性化したルールの結論部の結果得られた利益 (損失) によって評価できることになる。その評価に対応してルールが外部環境に適しているのかどうかという信頼度を与えるようにすればよい。以上のような

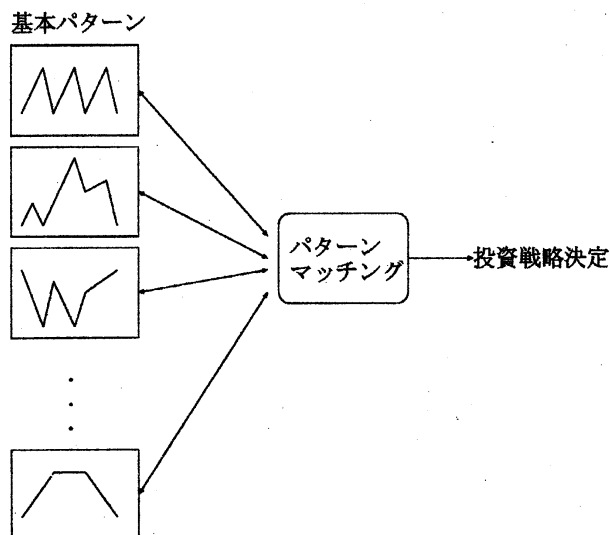


図 3: マッチングプロセス

評価を行なうためには、外部環境からメッセージ (過去の株価変動パターン) が送られて来た時、どのルールの条件部とマッチさせるかが重要となってくる。本システムでは個々のルールの条件部に与えられたパターンと実際の変動パターンに GA を用いたパターンマッチングを行ない、そのマッチしたルールが活性化されるとした (図3参照)。

### 2.3 GA を用いたパターンマッチング

適応的ルール生成システムによって作られた基本パターンと実際の株の変動パターンとのマッチングをさらに GA を用いて行なう。

まず、適応的ルール生成システムのルールの条件部のパターンを GA によって部分的に拡大縮小する。基本パターンの点を  $(p_i, q_i), i = 1, 2, \dots, n_p$  とする。そしてそれらの前の点との差を  $x_j, y_j, j = 1, 2, \dots, n_p - 1$  とすると、

$$\begin{aligned} x_j &= p_{j+1} - p_j, \\ y_j &= q_{j+1} - q_j. \end{aligned} \quad (2)$$

$$(3)$$

$x_j, y_j$  を GA によって拡大、縮小することによって、基本パターン全体の変形を行ない、変形率が小さいパターンほど評価が高くなるように、1 から変形率を引いた値の絶対値を実際の変動との 2 乗誤差とたしたものを経済価値として、マッチの度合をみる。 $x_j, y_j$  の拡大、縮小率をそれぞれ  $\alpha_j, \beta_j$  とすると、変形後の点の差

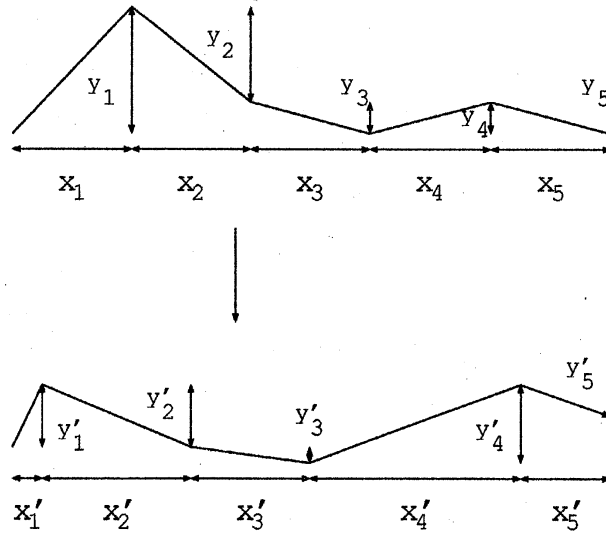


図 4: 基本パターンの変形

$x'_j, y'_j$  は、

$$\begin{aligned} x'_j &= \alpha_j x_j, \\ y'_j &= \beta_j y_j. \end{aligned} \quad (4)$$

$$(5)$$

となり (図 4 参照.), システムへの入力情報となる株価時系列を  $S(t), S(t), \dots, S(t+p)$ , とすると活性化されるルールは、

$$\begin{aligned} \operatorname{argmin}_n \quad & \sqrt{\sum_{j=1}^{n_p} (S(\sum_{n=1}^j x_n) - \sum_{k=1}^j y_k)^2} \\ & + \sum_{j=1}^{n_p} |1 - \alpha_j| + \sum_{j=1}^{n_p} |1 - \beta_j|. \end{aligned} \quad (6)$$

を満たすルール  $n$  となる。ここで、非整数時刻の株価は  $S(t)$  の線形補間によって求める。

### 3 遺伝的アルゴリズムによるルールの生成

#### 3.1 遺伝的アルゴリズム (GA)

遺伝的アルゴリズム (GA) とは, 生物進化のプロセスを模倣し, 問題に対する最適解を広域的に探索する確率的探索アルゴリズムである.

このアルゴリズムの大まかな手順としては, まず解の候補を染色体と呼ばれる遺伝子列で記述した. これはコーディングといわれるもので, GA においてもっとも重要な位置をしめている. その後, (1) 初期の個体集団をランダムに作成 (初期化) し, (2) それぞれの個体の環境への適応度の評価を行ない, (3) その適合度の高いものが次の世代により多く残るように選択する処理をし, (4) さらに交差, (5) 突然変異処理などにより新しい世代集団を構成していく. (2)~(5) の手順が繰り返され, さまざまな染色体をもつ個体を進化させもっとも適合する解候補を探索していく.

以下では, それぞれの手順の概要を示す.

##### 3.1.1 コーディング

遺伝的アルゴリズムにおいて, 染色体で定義される性質, 形質の外部表現が表現型とよばれ, 性質, 形質の染色体による内部表現を遺伝子型と呼ぶ. 例えば, そらまめの DNA による遺伝子配列表現が遺伝子型であり, その染色体によって実際に形づくられたそらまめ自体が表現型にあたる. 表現型から遺伝子型へ変換することをコーディングといい, 逆に, 遺伝子型から表現型への変換をデコーディングという. 染色体を構成する遺伝子記号には, 0, 1 の 2 値をとるバイナリ型, 整数型, 実数型, アルファベットなどが考えられる. どの遺伝子記号を用いればよいかは解こうとする最適問題に依存する.

適応的ルール生成システムでは, プロダクションシステムのルールを適切な遺伝子記号列で示す必要がある. ルールの条件部には, 過去の株価の変動パターンを与えなければならないので, 株価を 2 進数 (0, 1) で表現し, 取引戦略決定に用いる株価の期間分の記号列で表現する. それに対応するルールの結論部には条件部に対する取引内容を, 同じように 2 進数で表現した記号で表す.

$p$  期過去の値までを基本パターン表現に用い, 株価を  $b$  ビットで表すとすると, 1 つの染色体のビット数  $bits$  は

$$bits = bp + 1. \quad (7)$$

となる.

#### 3.2 初期化

コーディング過程で, 問題に対する解を染色体によってどのように表現するかを決定した後, 初期世代集団となる  $N$  個の染色体をランダムに生成する. 集団サイズ  $N$  は大きい方がより広域的探索が行なえるため, 大きい方がよいが計算時間の増大などの問題もあるので, そのことを考慮する必要がある.

本システムでは  $bits$  の長さをもつ記号列を  $N$  個発生させることになる.  $0 \sim 1$  の一様乱数を  $bits \times N$  個発生させ,  $0 \sim 0.5$  の範囲の値の場合は 1 を与え,  $0.5 \sim 1$  範囲の場合は 0 を対応させる.

#### 3.3 環境への適応度の評価

ここでは, 生成された個体のそれぞれについて環境, つまり最適化したい問題への適応度の計算を行なう. 適応度関数を  $f(x)$  とし, 個体  $j$  の表現型を  $x_j$  とすると,

$$fitness_j = f(x_j). \quad (8)$$

と求めることができる。

本システムでは、環境への適応度の評価を新たに計算する必要はなくそれぞれのルールに与えられた信頼度を環境への適応度とみなすことができる。

信頼度 (適応度) が高いルールほど、現在の株価変動特性をよりよく表現していることになる。

### 3.4 選択

次に、計算された適応度をもとに次世代の個体を生成するための個体を確率的に選択する。適応度の高いものほどより高い確率で選択される。適応度の高い個体の選択確率を高くすれば高くするほど解の収束は速くなる。しかし、そうすると、似たような遺伝子型をもった個体が多くなり、すぐれた部分的な遺伝子型をもっている可能性のある適応度の低い染色体が生き残れない可能性がでてきて、広域的な探索が行なわれない場合がある。逆に、適応度の高いものの選択確率を低くしすぎると、ランダム探索と変わらない結果になってしまう。

上記のような確率的選択では、もっとも適合した染色体をもった個体を淘汰してしまう可能性があるため、1 番適応度の高い個体が残らなかった場合に、無条件にこの染色体を残すエリート保存処理が行なわれることもある。

本システムでは、個々のルールは独立したものではなくルール全体で 1 つの取引パターンを形成しているので、ルール集合に大きな変更を加えると取引利益率の減少、遺伝的アルゴリズムにおける退化が起こる可能性がある。よって、適応度にあたる信頼度が比較的大きなものは優先して高確率で保存することとした。優先する個体は全体の 20% 程度とし、その他の個体については信頼度に応じた確率、

$$p(x_j) = \frac{f(x_j) - \min_i \{f(x_i)\}}{\sum_k (f(x_k) - \min_i \{f(x_i)\})} \quad (9)$$

によって選択を行なう。

### 3.5 交叉

選択により次世代を構成する個体集団が選ばれた後、その中から 2 つの個体を親とする新しい子の個体を染色体の組み替えによってつくり生成する。交叉方法には染色体の 1 点のみで交叉を行なう 1 点交叉法、複数ヶ所で行なう多点交叉法、ある任意のパターンに従い、どちらの親の遺伝子を継承するか決定する一様交叉法、それぞれの親の中間の値を継承させる混合交叉法などがある。どの交叉法を用いるかは、解こうとする最適問題、コーディングの方法に強く依存するので、親の形質が適切に継承されるように選ぶ必要がある。交叉により、選択で生き残った解として適切であろうルールパターンをもった染色体の性質を受け継ぎつつ、他の染色体の特質も受け継いだ新たな染色体ができ、親よりも優れた個体が生成される可能性がある。

本システムでは、選択された個体を確率  $P_{cross}$  によってランダムに選ばれたペアと多点 (2 点) で交叉を行なった。

交叉により親ルールの変動パターンを継承しつつ、新たな条件部、結論部をもったルールが生成される。このように新たに生成されたルールは条件部がマッチする外部環境からのメッセージがあった場合、その結論部を取引内容として示し、評価が行なわれ、外部の環境に適している場合は次の世代で選択され、新たなルールを形成していくことになる。

### 3.6 突然変異

突然変異オペレーションとは, ある個体の染色体の一部をある確率に従い書き換えるものである. 交叉と同様に, この突然変異の方法もいくつか種類があり, 主なものとしては染色体の一部を同じ染色体の他の部分に移す転座法, 染色体上である一定の長さの遺伝子を重複させる重複法, 染色体の部分的な遺伝子配列の順序を入れ換える逆位法, 染色体の一部を確率的に違う値に書き換える反転法などがある.

突然変異では, 最適解探索のランダムな部分を与えており, 特定の染色体パターンに収束することを防ぐことを目的としている.

本システムでは, それぞれの個体にある確率  $P_{mut}$  で突然変異オペレーションを行ない, 染色体の一部を違う値に書き換える. 染色体は 0,1 のバイナリー値をとるので, 0 なら 1, 1 なら 0 へ書き換えることになる.

## 4 取引シミュレーション

本システムによる株取引のシミュレーションを行なう.

適応的ルール生成システム (ARGS) のパラメーターとしては, ルール集団を構成するバイナリ値表現されたルールの個数は 1000 とし, 10 進数で表現された株価を 2 進数に変換する場合のビット数は 10 ビットとした. よって, 1 つのルールを表現する染色体の遺伝子記号列は全体で  $10 \times p + 1$  ビットとなる. 学習データ数 100, 最大 GA 世代数 100, パターンマッチに用いる時系列のサンプル数 10, 基本パターンのサンプル数 5, 交叉確率 0.2, 突然変異確率 0.001 とした. また, これらのルールは遺伝的アルゴリズムによって, その信頼度に従って新たなルール集団を構成することになる.

パターンマッチのパラメーターとしては, 個体数 100, 最大世代数 30, 交叉確率 0.7, 突然変異確率 0.001 とし, シミュレーションを行なった. 図 5 から図 7 にはパターンマッチングの例として, 基本テンプレート, 時系列の区間をずらしていった時のマッチング誤差, 変形後のテンプレートと区分時系列を示している.

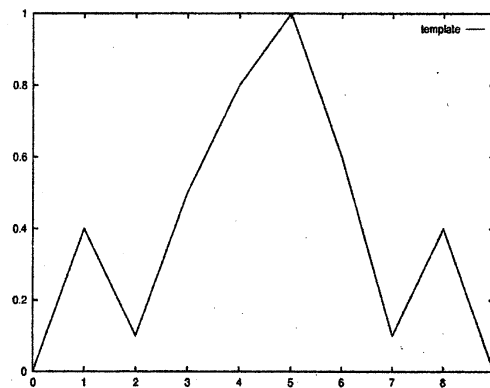


図 5: テンプレート (head & shoulders)

また, 利益率の定義は, 現時点で買った株を step 日後に決済する買いシグナルによる利益率は,

$$R_b = \sum_{n=1}^{N_b} \frac{S(n + \text{step}) - S(n)}{\text{step} N_b}. \quad (10)$$



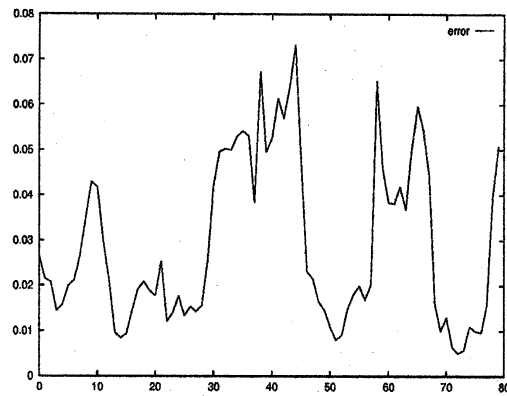


図 6: マッチング誤差

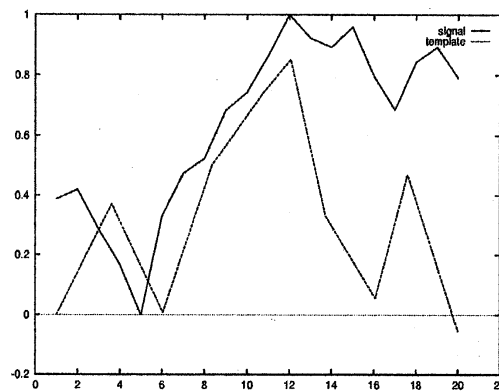


図 7: 変形パターンと原時系列

売りシグナルによる step 先取引の利益率は,

$$R_s = \sum_{n=1}^{N_s} \frac{S(n) - S(n + nstep)}{step N_s}. \quad (11)$$

step 先取引の平均利益率は,

$$R(step) = \frac{R_b + R_s}{2}. \quad (12)$$

ここで,  $S(n)$  は  $n$  時点における株価,  $N_b, N_s$  はそれぞれ売り買いの取引回数である.

比較モデル (AR) の平均利益率は,

$$\begin{aligned} & \text{if } pred(n + step) > S(n) \\ & \quad \text{then } R_{ar} = \frac{S(n+step) - S(n)}{step N_{ar}}, \\ & \text{if } pred(n + step) < S(n) \\ & \quad \text{then } R_{ar} = \frac{S(n) - S(n+step)}{step N_{ar}}. \end{aligned}$$

とする.  $pred(n)$  は  $n$  時点の株価の予測値,  $N_{ar}$  は取引回数である.

表 1 には本システム (ARGS) と AR モデルによる投資モデルの平均利益率をまとめている.

	step=1	2	3	4	5	6	7	8	9	10
ARGS	22.804	21.557	22.227	23.019	23.703	23.801	23.836	23.910	23.833	23.699
AR	-3.714	-18.237	-16.231	-9.500	-9.534	-7.662	-3.199	1.860	5.351	6.110

表 1: 取引平均利益率

## 5 むすび

本論文ではクラシファイアシステムの枠組を利用し、新しい株価投資支援システムである適応的ルール生成システムを構築し、AR モデルとの投資利益率の比較を行なった。

今後の課題としては、本システムにカオス、フラクタル理論を取り入れたシステムを考察し、このようなシステムによる取引主体が多数集まった場合の株価市場の構成過程を実際の株価市場との比較、検討することが考えられる。

## 参考文献

- [1] 北野宏明編: " 遺伝的アルゴリズム ", 産業図書, 1993.
- [2] 北野宏明編: " 遺伝的アルゴリズム 2 ", 産業図書, 1993.
- [3] Belew ,R.: " Artificial Life : A Constructive Lower Bound for Artificial Intelligence ", IEEE Expert, Vol.6, No.1, Feb. 1991.
- [4] Fogel, L., Owens, A. and Walsh, M: " Artificial intelligence through simulated evolution ", New York, John Wiley, 1966.
- [5] Goldberg, D.: " Computer-aided gas pipeline operation using genetic algorithms and rule learning ", Ph.D Thesis, University of Michigan, 1983.
- [6] Goldberg, D.: " Genetic Algorithms in Search, Optimization, and Machine Learning ", Addison-Wesley, 1989.
- [7] Langton, C.: " Artificial Life ", Addison Wesley, 1989.
- [8] Powell, D., Tong, S., Skolnick, M.: " EnGENEous: Domain independent ,machine learning for design optimization ", Proc. of ICGA-89, 1989.
- [9] Robertson, G.: " Parallel Implementation of Genetic Algorithms in a Classifier System ", Davis, L. (Ed.), Genetic Algorithms and Simulated Annealing, Morgan Kaufmann Publishers, 1987.
- [10] Smith, S.F: " A Learning System Based on Genetic Adaptive Algorithms ", Ph.D.Thesis, University of Pttsburgh, 1980.
- [11] Tanese, R.: " Distributed Genetic Algorithms ", Proc. of ICGA-89, 1989.